

Łukasz DZIERŻANOWSKI*
Robert KOZA*

IDENTYFIKACJA OBIEKTÓW NA PODSTAWIE CHMURY PUNKTÓW

Artykuł przedstawia problematykę identyfikacji obiektów w obrazach uzyskanych ze skanerów przestrzennych z wykorzystaniem chmury punktów. Przybliża zasadę algorytmu ICP, popularnej metody wykorzystywanej w celu dopasowania do siebie dwóch chmur punktów w oparciu o wskazanie najbliższego sąsiada. Ponadto zaprezentowano metodę kd-drzew znacznie zwiększającą wydajność algorytmu ICP, co zostało poparte wynikami działania autorskiej aplikacji służącej do dopasowywania chmur punktów z wykorzystaniem różnych metod: kd-drzew, triangulacji Delaunay oraz metody Brute-Force.

1. WPROWADZENIE

Skanowanie przestrzenne jest technologią, która stale zyskuje na popularności, znajdując zastosowanie w różnych systemach diagnostyczno-remontowych. Pozwala ono bowiem na zdalne wykonanie analizy uszkodzeń, detekcję zmian geometrii itd. Szybkość i kompleksowość równocześnie wykonywanych pomiarów sprawia, że jest to technologia szczególnie wartościowa dla zastosowaniach związanych z czynnościami remontowo-eksploatacyjnymi w obszarze systemu elektroenergetycznego, z względu na wysoki koszt czasu wyłączeń związanych pracami diagnostycznymi.

Ze względu na możliwość pojawienia się różnych konfiguracji terenu, przeszkód lub złożoności samej infrastruktury, przez okluzje i ograniczony zasięg sensorów, wiele standardowych środowisk aplikacyjnych dedykowanych skanerom przestrzennym, może wymagać dokładnych metod łączenia kilku chmur punktów w jeden. Dopasowanie geometryczne długo pozostawało wciąż wąskim gardłem dla systemów wizji komputerowej i lokalizacji mobilnych robotów, aczkolwiek ostatnie lata przyniosły postęp dla pewnych szczególnych zastosowań [2, 3, 4]. Problemem w rozpoznawaniu obiektów jest dopasowanie zaobserwowanych danych do wcześniej zapisanego modelu reprezentującego różne przedmioty zainteresowania. Pierwotnie wykorzystywano w tym celu algorytmy wykorzystujące podstawowe figury geometryczne takie jak punkty,

* Politechnika Opolska.

linie i figury płaskie, jednak ze względu na duże ograniczenia, wraz z rozwojem możliwości sprzętu, zaczęto stosować chmury punktów w oparciu o algorytm ICP, który stale poddawany istotnym modyfikacjom zwiększającym jego dokładność.

2. ALGORYTMY WSPOMAGAJĄCE INTEGRACJĘ CHMUR PUNKTÓW

2.1. Działanie Algorytmu ICP

Optymalna rejestracja chmur punktów jest istotnym problemem, dlatego w celu eliminacji błędów lub wypełnienia dziur informacyjnych, stosuje się transformacji dla jednej z chmur i możliwe dobre dopasowanie jej do drugiej chmury. Pierwszy zbiór punktów nazwano punktami modelu a drugi punktami danych. Typowe podejście polega na zastosowaniu transformacji dla punktów danych w celu jak najlepszego dopasowania ich do punktów modelu. Przyjęto, że następujące oznaczenia dla zbioru punktów modelu:

$$M = \{m_1, m_2, m_3, \dots, m_N\} \quad (1)$$

oraz dla zbioru punktów danych:

$$D = \{d_1, d_2, d_3, \dots, d_N\} \quad (2)$$

W celu sformułowania matematycznie dokładności dopasowania należy zdefiniować błąd systemu metrycznego lub funkcję celu E. Popularną miarą jest suma kwadratów błędów – są to kwadraty odległości od punktów w jednej chmurze do ich najbliższych sąsiadów w drugiej po zastosowaniu transformacji τ .

$$E = \sum_{i=1}^N \| \tau(d_i) - m_i \|^2 \quad (3)$$

W przestrzeni 3-D, taka transformacja posiada sześć stopni swobody - trzy obroty i trzy przesunięcia - funkcja celu staje się funkcją sześciu zmiennych. Można teraz sformułować wyrażanie obrotu w zakresie obrotu macierzy R oraz przesunięcia o wektor \vec{T} .

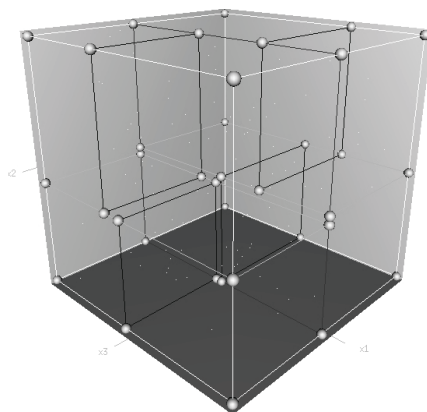
$$E = \sum_{i=1}^N \| R d_i + \vec{T} - m_i \|^2 \quad (4)$$

W powyższej analizie nie jest znana informacja o wzajemnej zależności pomiędzy punktami, co oznacza że nie wiadomo, który punkt m_i będzie najbliższym sąsiadem punktu d_i po zastosowaniu transformacji. Ta informacja może jednak zostać oszacowana w oparciu o dane o „najbliższym sąsiadzie” punktu sprzed transformacji. Zostało to wykorzystane w algorytmie ICP (ang. Iterative Closest Point – Iteracyjny Najbliższy Punkt). Metody minimalizacji zaprezentowane w dalszej części artykułu, wykorzystują wybierając niektóre operacje transformacji i określają jak powinny się zmieniać aby zmniejszyć wartość funkcji E.

2.2. Zastosowanie drzewa k-wymiarowego

Algorytm ICP w swojej oryginalnej formie jest niewydajny, aby móc go zastosować bezpośrednio w aplikacjach, gdyż czas wykonywania obliczeń rośnie kwadratowo wraz ze wzrostem ilości punktów. Istnieje wiele sposobów przyspieszenia obliczeń, jednym z nich może być ograniczenie liczby punktów. Metodę tę jak i kilka innych opisano szerzej w artykule [6]. Wszystkie z wymienionych metod zakładają istnienie siatki punktów danych wejściowych, które powinny znajdować się w prostokątnych macierzach nie zawierających pustych elementów. W rzeczywistych danych znajduje się jednak szum gaussowski i uzyskanie siatki nie jest możliwe lub bardzo utrudnione. Aby poradzić sobie z surowymi danymi wykorzystano strukturę danych w postaci drzewa. Drzewa k-wymiarowe służą do podziału wielowymiarowej przestrzeni na podprzestrzenie w taki sposób, by wyszukiwanie najbliższych sąsiadów było jak najszybsze.

Drzewo-kd jest drzewem binarnym, gdzie w każdym węźle znajduje się k-wymiarowy punkt. Każdy węzeł, który nie jest liściem tworzy hiperpłaszczyznę podziału, która dzieli przestrzeń na dwie podprzestrzenie. Punkty po lewej stronie hiperpłaszczyzny reprezentują lewe poddrzewo zaczynające się w tym węźle a prawe punkty, prawe poddrzewo. Kierunek hiperpłaszczyzny jest wybierany zgodnie z wektorem normalnym płaszczyzny. Najczęstszym wyborem miejsca podziału jest mediana zbioru. Takie podejście powoduje utworzenie dobrze zbalansowanego drzewa i zwiększa wydajność wyszukiwania najbliższych sąsiadów. Rys. 1 przedstawia przykładowe kd-drzewo. Pierwszy podział (czerwony) dzieli główny zbiór (biały) na dwa podzbiory. Każdy z nich jest następnie dzielony (zielony) na kolejne podzbiory. A te dzielone na kolejne podzbiory (niebieski). Gdy już nie można dalej dzielić, każdy z ośmiu zbiorów jest tzw. Liściem, tj. elementem, który dalej nie podlega rozgałęzieniom. Zastosowanie tablicy z wcześniej znalezionymi najbliższymi sąsiadami dodatkowo przyspiesza działanie algorytmu ICP [8].

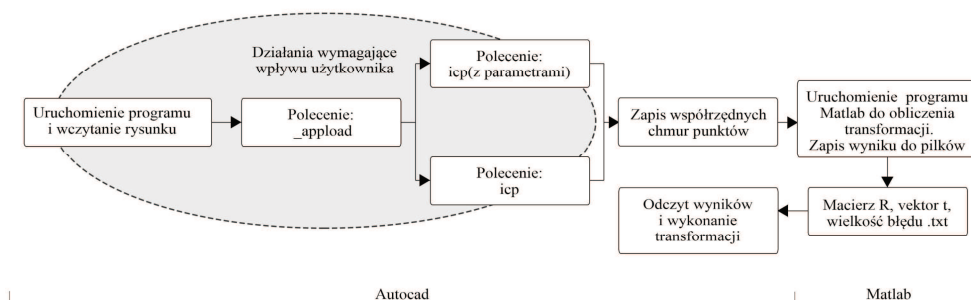


Rys. 1. Przykład struktury drzewa kd [7]

Dla n -elementowego zbioru punktów P uzyskuje się wtedy drzewo rozmiaru $O(n)$, które można skonstruować w czasie $O(n \log n)$. Czas odpowiedzi na zapytanie o punkty należące do prostokąta R wynosi $O(k+n^{(d-1)/d})$, gdzie k jest liczbą znalezionych punktów, a d – wymiarem przestrzeni

3. PROJEKT APLIKACJI

Celem powstania aplikacji było umożliwienie rejestracji chmur punktów wewnątrz środowiska AutoCAD. Założono że oba dopasowywane zbiory punktów są dostępne i nie ma potrzeby ich generowania z modelu. Algorytm, którego użyto składa się z kilku etapów. Stworzona aplikacja jest programem wykonującym wiele złożonych obliczeń, które nie mogły zostać wykonane w środowisku CAD w tym celu utworzono skrypt, który za pomocą programu Matlab dokonuje obliczeń i zwraca wynik do AutoCAD'a gdzie wykonywany jest ostateczny etap czyli transformacja jednej z chmur. Jednym z założeń wstępnych było uzyskanie dopasowania dla dużych chmur punktów. Aby spełnić następujące wymaganie należało znaleźć wydajną strukturę danych czyli omawiane wcześniej drzewo k -wymiarowe. Stworzona aplikacja umożliwia porównanie działania algorytmów nie tylko na podstawie subiektywnej oceny wizualnego efektu działania, ale również w oparciu o porównania określonych deterministycznych współczynników. Gotowa aplikacja została tak skonstruowana, aby użytkownik miał możliwość łatwego wyboru algorytmów i jego parametrów, które mają zostać zastosowane do rejestracji chmur punktów.

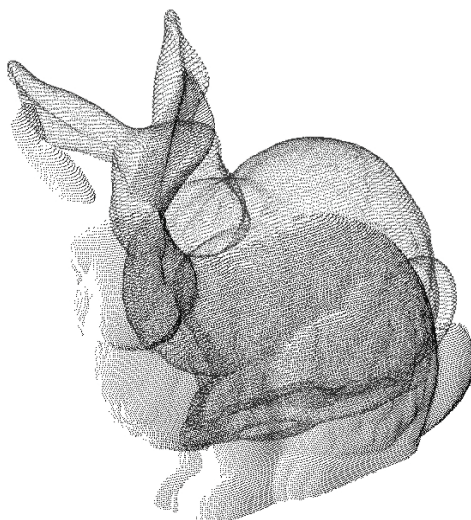


Rys. 2. Schemat działania programu

Schemat działania aplikacji został sprowadzony do trzech kroków: uruchomienia środowiska AutoCAD i wczytania rysunku zawierającego chmurę punktów (na odpowiednio nazwanych warstwach), załadowania aplikacji oraz wyboru algorytmu i jego parametrów. Program działa z linii poleceń AutoCAD'a,

użytkownik otrzymuje informacje o tym jakie są dostępne polecenia zaraz po wczytaniu aplikacji.

Scena na której przetestowano aplikację przedstawia model królika zeskanowanego przez naukowców z Uniwersytetu Stanford [5]. Jest on popularnym obiektem testów różnych algorytmów przetwarzania obiektów 3D. Przygotowana scena testowa „StanfordBunny” została złożona z dwóch chmur o łącznej liczbie ~76000 punktów.



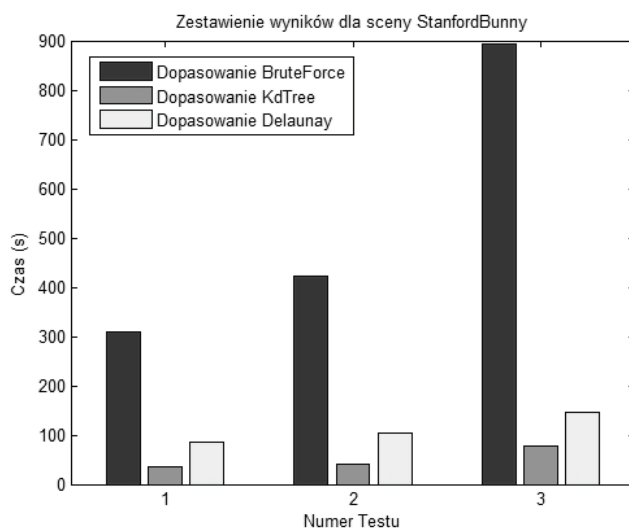
Rys. 3. Skany prezentujące obiekty

Zaprojektowana aplikacja została wyposażona w dwie dodatkowe metody wyszukiwania najbliższych par punktów w celu porównania wzrostu czasu dla zaproponowanej struktury drzewa k-wymiarowego. Są to triangulacja Delaunay polegająca na podziale obszaru na trójkąty, gdzie każde koło opisane jest na trzech sąsiadujących ze sobą punktach z tessalacji Voronoia nie zawiera żadnego innego punktu oraz metoda Brutalnej Siły (BruteForce) polegająca na porównywaniu każdego punktu z obu zbiorów. Każdą z metod testowano trzykrotnie, wywołując funkcję dopasowania najpierw bez parametrów, z parametrem ekstrapolacji funkcji błędu oraz z parametrem odrzucania 10% najgorszych par – najbardziej oddalonych od siebie.

Badania przeprowadzone na scenie „StanfordBunny” potwierdzają, że metoda dopasowania najbliższych par punktów ma największy wpływ na czas zbieżności chmur punktów. Wielkość ostatecznego błędu RMS jest taka sama w przypadku stosowania jednakowych parametrów dla różnych metod dopasowania. Kolejny rysunek prezentuje zestawienie wyników testu dla trzech zaimplementowanych metod dopasowania.

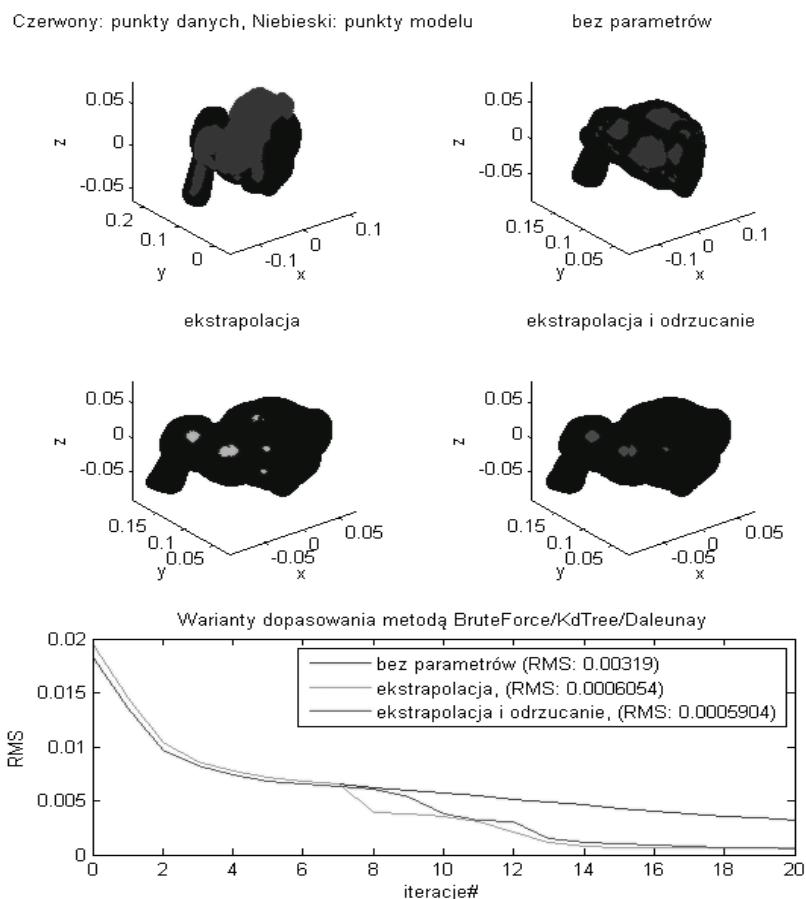
Tabela. 1. Zestawienie wyników dla sceny „StanfordBunny”

Lp.	Metoda dopasowania	Czas (sek.)	Liczba iteracji	Błąd RMS
1.	2	3	4	5
1	BruteForce	310,07	20	0,003189
2	KdTree	35,83	20	0,003189
3	Delaunay	86,16	20	0,003189
4	BruteForce z ekstrapolacją	421,88	20	0,000605
5	KdTree z ekstrapolacją	41,75	20	0,000605
6	Delaunay z ekstrapolacją	104,1	20	0,000605
7	BruteForce z ekstrapolacją i odrzucaniem	892,43	20	0,000542
8	KdTree z ekstrapolacją i odrzucaniem	77,58	20	0,000542
9	Delaunay z ekstrapolacją i odrzucaniem	145,72	20	0,000542



Rys. 4. Zestawienie wyników czasu wykonywania dopasowania dla sceny „StanfordBunny”

Wykonane testy pokazują, że dla chmur punktów pozbawionych szumu i stosunkowo dobremu dopasowaniu początkowemu otrzymanemu dzięki manualnej transformacji, aplikacja jest w stanie uzyskać dopasowanie na poziomie 0,000605 już po około 15 krokach iteracji z wykorzystaniem ekstrapolacji. Dodatkowo stosując drzewa kd można osiągnąć wspomniany wyżej poziom zbieżności już po około 41 sekundach (metoda iteracyjna daje ponad 420 sekund). Ten przykład jest interesujący również dlatego, że pokrycie pomiędzy tymi chmurami wynosi ~40 – 50%.



Rys. 5. Przebieg testu dla sceny „StanfordBunny”

Po przeprowadzeniu wszystkich testów można zauważyć pewną prawidłowość. Wyszukiwanie najbliższych par punktów metodą brutalnej siły jest od 8 do 10 razy wolniejsze od najszybszej metody drzewa k-wymiarowego, które z kolei jest dwukrotnie szybsze od metody triangulacji Delaunay. Kolejnym faktem wartym odnotowania jest to, że dla chmur punktów bez szumu wywołanie funkcji dopasowania z parametrem odrzucania najgorszych par punktów tylko nieznacznie poprawia efekt końcowego dopasowania. W większości przypadków wybór metody kd-drzew i ekstrapolacji funkcji dopasowania będzie najlepszym rozwiązaniem w stosunku wielkości błędu RMS do czasu wykonywania. Badania potwierdzają, że zastosowanie kd drzew dla trzech wymiarów jest skutecznym sposobem przyspieszenia algorytmu ICP.

4. PODSUMOWANIE

Wykonanie aplikacji pozwoliło na przeprowadzenie testów porównujących efektywność i dokładność różnych metod dopasowania dwóch chmur punktów w taki sposób, by razem tworzyły pełny obraz zeskanowanego modelu.

Algorytm kd drzewa wykorzystujący najbliższe pary korespondujących ze sobą punktów okazał się najlepszy z wszystkich zaimplementowanych. Umiarkowanie długi czas dopasowania, nie wzrastający zbyt szybko przy zastosowaniu dodatkowych funkcji takich jak ekstrapolacja, czy odrzucanie najgorszych par sąsiadujących punktów nawet przy zwiększeniu ilości danych wejściowych, pozwala uzyskać znakomitą jakość dopasowania chmur punktów. Algorytm triangulacji Delaunay to zaraz po drzewie kd najszybsza metoda, jej osiągi pozwalają na stosunkowo szybką rejestrację chmur punktów. Zaimplementowana metoda brutalnej siły pozwala na porównanie zależności czasu wykonywania obliczeń dla wymienionych wcześniej algorytmów wyszukiwania najbliższych (w domyśle korespondujących) par punktów.

LITERATURA

- [1] SEGAL A., HAEHNEL D., THRUN S.: Generalized-icp, Proceedings of Robotics: Science and Systems, Seattle, USA 2009.
- [2] BESL P.J., ND. MCKAY: A Method for Registration 3D Shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14(2) ss. 239-256.
- [3] RUSINKIEWICZ S., LEVOY M.: Efficient Variants of the ICP Algorithm, Proc. 3DIM, 2001 ss. 145-152.
- [4] ZYCHEWICZ A., SIEMIĄTKOWSKA B.: Zastosowanie algorytmów ICP i SIFT w lokalizacji robotów mobilnych, *Pomiary Automatyka i Robotyka* 2/2010 ss. 297-303.
- [5] Stanford University, <ftp://graphics.stanford.edu/pub/3Dscanrep/bunny.tar.gz>
- [6] Rusinkiewicz S.: Real – Time Acquisition and Rendering of Large 3D Models. Ph.D. Dissertation, Stanford University, 2001
- [7] http://pl.wikipedia.org/wiki/Drzewo_kd
- [8] Siemiątkowska B., Gnatowski M., Zychewicz A., *Zastosowanie algorytmów ICP i SIFT w lokalizacji robotów mobilnych*, *POMIARY AUTOMATYKA KONTROLA*, **3**, 275-278, 2010.

OBJECTS IDENTIFICATION BASED ON A POINT CLOUD

The paper presents the aspects of objects identification based on 3D scanning with use of the point cloudes. The ICP, a popular algorithm for point cloud adjustment based on the nearest neighbor iteration. Moreover a kd-tree method for ICP efficiency improving has been described along with the results of testing the authors' program for point cloud adjustment based on kd-tree, Delaunay's triangulation and brute force methods.